Interspeech 2022
18-22 September 2022, Incheon, Korea

Dual Path Embedding Learning for Speaker Verification with Triplet Attention

Bei Liu, Zhengyang Chen, Yanmin Qian[†]

MoE Key Lab of Artificial Intelligence, AI Institute
X-LANCE Lab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{beiliu, zhengyang.chen, yanminqian}@sjtu.edu.cn

Abstract

Currently, many different network architectures have been explored in speaker verification, including time-delay neural network (TDNN), convolutional neural network (CNN), transformer and multi-layer perceptrons (MLP). However, hybrid networks with diverse structures are rarely investigated. In this paper, we present a novel and effective dual path embedding learning framework, named **D**ual **P**ath **Net**work (DPNet), for speaker verification with triplet attention. A new topology of integrating CNN with a separate recurrent layer connection path internally is designed, which introduces the sequential structure along depth into CNN. This new architecture inherits both advantages of residual and recurrent networks, enabling better feature re-usage and re-exploitation. Additionally, an efficient triplet attention module is utilized to capture cross-dimension interactions between features. The experimental results conducted on Voxceleb dataset show that our proposed hybrid network with triplet attention can outperform the corresponding ResNet by a significant margin.

Index Terms: speaker verification, dual path embedding learning, triplet attention

1. Introduction

The task of speaker verification (SV) is to verify the speakers' identities by utilizing voice as the biometric feature. In recent years, the paradigm of state-of-the-art SV systems has shifted from i-vector [1] combined with probabilistic linear discriminant analysis (PLDA) [2] towards deep speaker embedding learning method [3, 4], where deep neural networks (DNN) are utilzied to take the frame-level features of an utterance as input and directly produce an utterance level representation as speaker embeddings for similarity measurement. These embeddings are obtained via the pooling mechanism in which mean and standard deviation are generally calculated. DNN-based SV systems can be effectively trained by multi-class classification, where softmax [5] or AAM-softmax [6] can be adopted as loss function. Subsequently, the extracted embeddings are used in a standard backend, e.g. cosine similarity calculation.

According to the network architecture, DNN-based SV systems proposed in previous works can be divided into four different types: TDNN-based [5, 7, 8, 9, 10], CNN-based [11, 12], transformer-based [13, 14] and MLP-based [15]. Time delay neural network (TDNN) is known as the ability to learn the temporal dynamics of the signal with wide context, which adopts a hierarchical and incremental architecture to process different temporal resolution [16]. These characteristics make TDNN naturally suitable for speech tasks. [5] firstly utilizes a TDNN architecture with a multiclass cross entropy objective for text-independent speaker verification. x-vector [7] and its descendants [8, 9] are further proposed to improve the performance. ECAPA-TDNN [10] obtains astounding results by making multiple architectural enhancements to the x-vector. For CNN-based SV systems, [11] introduces ResNet [17] as the speaker embedding extractor in VoxSRC 2019 for the first time. In addition, [13] presents a transformer-based system with self-attention encoder and pooling layer to obtain a discriminative speaker embedding, which is inspired by transformer's effectiveness in natural language processing and computer vision [18, 19]. [14] makes further efforts to improve the transformer-based system by strengthening local information modeling. Plus, [15] attempts to build a pure MLP network without convolution or self-attention, which shows competitive results. However, the existing DNN-based SV systems mostly focus on single network structure. Hybrid networks with diverse structures have been rarely discussed in SV task, which demonstrate the superiority in other fields [20, 21, 22, 23, 24, 25].

In this paper, we design a novel hybrid network structure, namely **D**ual **P**ath **Net**work (DPNet), for speaker verification with triplet attention. Compared to previous works, the proposed DPNet consists of two paths: residual path and recurrent path. It integrates CNN with sequential information flow via recurrent layer connection along depth. This new architecture can enjoy the benefit of better re-using information from previous layers. Moreover, an efficient triplet attention module is introduced to model cross-dimension interactions between features. Experiments conducted on Voxceleb [26, 27] demonstrate that our proposed DPNet with triplet attention can outperform the corresponding ResNet by a large margin.

2. Related Work

Hybrid Networks: Hybrid networks generally integrate different network structures together, which show the superiority over single network structure in various fields [20, 21, 22, 23, 24, 25]. Several hybrid variants have been extensively studied in recent years, including CNN-CNN, CNN-RNN and CNN-Transformer. [20] designs a CNN-CNN hybrid network combining ResNet with DenseNet, which achieves better performance than state-of-the-arts. For CNN-RNN hybrids, [21] augments convolutional residual networks with a long short term memory mechanism for image classification. [22] adpots RNN to process the outputs of CNN for visual description. [23] builds a convolutional LSTM model for the precipitation nowcasting problem. Lately, transformer has taken computer vision by storm and CNN-Transformer hybrids emerge subsequently.

[†] corresponding author

10.21437/Interspeech.2022-481

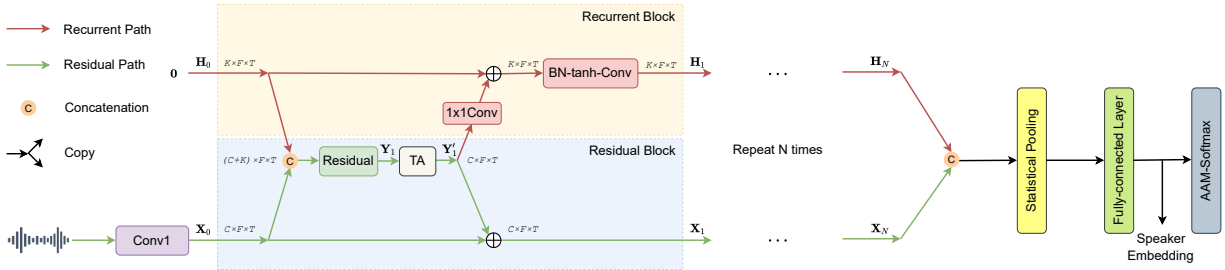

Figure 1: *The topology of dual path embedding learning framework. Two parallel paths exist where information exchange and fusion happen. Finally, the features from two paths are concatenated to obtain speaker embedding.* ***Residual path:*** *normal residual learning with residual blocks.* ***Recurrent path:*** *this is a recurrent path along depth which is equivalent to an unfolded RNN where* $\mathbf{H}_i$ *is the hidden state. It can accumulate information from previous layers via recurrent blocks and exchange with the residual path.* ***TA:*** *triplet attention module inserted after the residual mapping in each residual block.*

Table 1: *The structure of DPNet34. It consists of the residual and recurrent path. Similarly to ResNet34, the architecture is divided into four stages, which contains 3, 4, 6, 3 blocks individually.*

Stage	Residual Path	Recurrent Path
–	[Conv-BN-ReLU]	**0**
1	[Conv-BN-ReLU Conv-BN] × 3	[Conv BN-tanh-Conv] × 3
2	[Conv-BN-ReLU Conv-BN] × 4	[Conv BN-tanh-Conv] × 4
3	[Conv-BN-ReLU Conv-BN] × 6	[Conv BN-tanh-Conv] × 6
4	[Conv-BN-ReLU Conv-BN] × 3	[Conv BN-tanh-Conv] × 3

[24] proposes Conformer to take advantage of convolutional operations and self-attention mechanisms for enhanced representation learning. [25] proves that self-attention and convolution are complementary, and designs AlterNet to combine them together. In this paper, we introduce a novel hybrid network structure for speaker verification task to improve the representation capabilities of SV systems.

Attention Modules in SV: Attention modules have been broadly applied in DNN-based SV systems [28, 29, 30, 31]. [28] incorporates squeeze-and-excitation module into ResNet. [29] proposes convolutional attention for modelling temporal and frequency information independently. [30] adopts duality temporal-channel-frequency attention. [31] utilizes simple attention module. This work employs an efficient triplet attention module which can be integrated with DPNet seamlessly.

3. Proposed Method

In this section, we describe the proposed dual path embedding learning framework along with triplet attention module in detail.

3.1. Dual Path Embedding Learning Framework

In the proposed dual path embedding learning framework, two paths exist: residual path and recurrent path. The residual path is from the commonly-used ResNet. The recurrent path is elaborately designed to provide the current layer with the accumulation of previous layers' information via recurrent connection along depth [32] . Finally, the resulting features of the two paths are aggregated to obtain speaker embedding. Fig. 1 schematically depicts the overall topology of this framework. Take DPNet34 as an example, the structural details are presented in Table 1.

Residual path: We adopt ResNet18 and ResNet34 as the residual path in this work. Assume that there are N residual blocks in total, the feature map of i-th residual block can be denoted as $\mathbf{X}_i \in \mathbb{R}^{C\times F\times T}$ where C, F and T represent the channel, frequency and time dimension respectively, for $1 \leq i \leq N$. These features are utilized to exchange information between the residual and recurrent path.

Recurrent path: In the recurrent path along depth, there exists one corresponding recurrent block for each residual block, which aims to accumulate the layer history and exchange information for better feature re-usage and re-exploitation in a sequential manner. We represent the i-th recurrent block feature map as $\mathbf{H}_i \in \mathbb{R}^{K\times F\times T}$ where K means the channel number in the recurrent block, for $1 \leq i \leq N$. In the experiments, K is set to 32. At the i-th step, the calculation process of residual path is as follows:

$$\mathbf{Y}_i = \text{Residual}([\mathbf{X}_{i-1} \cdot \mathbf{H}_{i-1}]) \tag{1}$$

$$\mathbf{X}_i = \mathbf{Y}_i + \mathbf{X}_{i-1} \tag{2}$$

where $\mathbf{X}_{i-1}$ and $\mathbf{H}_{i-1}$ are the output of previous residual block and recurrent block respectively. $[\cdot]$ stands for the concatenation along the channel dimension. Residual means normal residual learning. Specifically, inputs $\mathbf{X}_{i-1}$ and $\mathbf{H}_{i-1}$ are firstly combined via concatenation and then the results are passed to a residual block to get $\mathbf{Y}_i$. The residual mapping $\mathbf{Y}_i$ is finally added to the original input $\mathbf{X}_{i-1}$ to obtain the output of i-th residual block $\mathbf{X}_i$.

For the recurrent path along depth, it is equivalent to an unfolded RNN where $\mathbf{H}_i$ is the hidden state at step i, initialized as $\mathbf{0}$ at step 0. For the i-th step update, its input is the residual block output $\mathbf{Y}_i$ and previous hidden state $\mathbf{H}_{i-1}$.

$$\mathbf{H}_0 = \mathbf{0} \tag{3}$$

$$\mathbf{H}_i = \text{Conv2}(\tanh(\mathcal{B}(\text{Conv1}(\mathbf{Y}_i) + \mathbf{H}_{i-1}))) \tag{4}$$

where Conv1 is a point-wise convolution with output channel sizes of K. $\mathcal{B}$ stands for BatchNorm. tanh is the non-linear function. Conv2 indicates a 3x3 convolution. Specifically, the residual mapping $\mathbf{Y}_i$ is firstly compressed in channel dimension via 1x1 convolution and then added to $\mathbf{H}_{i-1}$. Subsequently, the resulting feature is processed through a batch normalization, a tanh activation and a 3x3 convolution. The point-wise convolution and 3x3 convolution are shared across all the recurrent blocks similar to RNN.

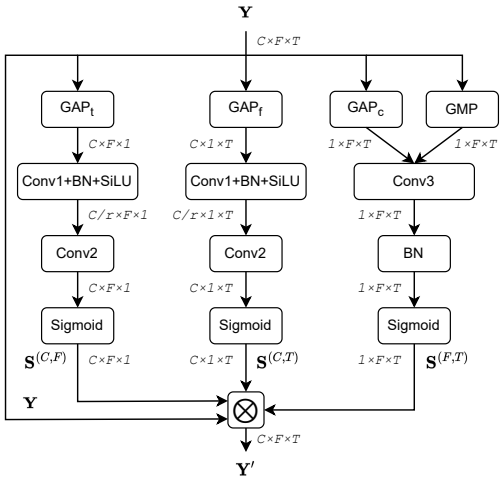

Figure 2: *Illustration of TA.*

From the above statements, we can see that our dual path architecture is not a simple combination of CNN and RNN. The key idea is the accumulation and exchange of information between two paths, which is crucial for enhanced feature learning.

Embedding layer: To obtain speaker embedding, the outputs of the last residual and recurrent block $\mathbf{X}_N, \mathbf{H}_N$ are firstly concatenated along the channel dimension. Then the resulting features are fed into a statistical pooling layer [33] to map the variable-length representation into a low dimension vector, which is then transformed by a full-connected (FC) layer to generate speaker embedding $\mathbf{e}$. The calculation is presented below:

$$\mathbf{Z} = [\mathbf{X}_N \cdot \mathbf{H}_N] \tag{5}$$

$$\mathbf{e} = \text{FC}(\mathcal{P}(\mathbf{Z})) \tag{6}$$

where $\mathcal{P}$ means the statistical pooling layer. FC is the full-connected layer.

3.2. Triplet Attention Module

Inspired by [29, 30], we design a novel and efficient attention module, namely triplet attention (TA), for speaker verification. Different from previous attention mechanisms, cross-dimension interactions are introduced into our proposed TA by dedicating three separate branches to capture the inter-dependencies between the channel and frequency dimension (C, F), the channel and temporal dimension (C, T), the frequency and temporal dimension (F, T) respectively. Fig. 2 is the overview of our proposed methods.

As Fig. 2 shows, given an input feature $\mathbf{Y} \in \mathbb{R}^{C\times F\times T}$, the first branch builds the interactions between the channel and frequency dimension (C, F). The resultant attention map $\mathbf{S}^{(C,F)} \in \mathbb{R}^{C\times F\times 1}$ is generated via:

$$\mathbf{S}^{(C,F)} = \sigma(\text{Conv2}(\text{SiLU}(\mathcal{B}(\text{Conv1}(\text{GAP}_\text{t}(\mathbf{Y})))))) \tag{7}$$

where Conv1 and Conv2 are point-wise convolution with output channel sizes of C/r and C respectively. r is the channel reduction ratio. $\mathcal{B}$ stands for BatchNorm. GAP_t is one-dimensional global average pooling along the temporal dimension. SiLU [34] is the non-linear function. σ is the sigmoid function.

Similarly, in the second branch, the attention map between the channel and temporal dimension (C, T), $\mathbf{S}^{(C,T)} \in \mathbb{R}^{C\times 1\times T}$, is obtained by:

$$\mathbf{S}^{(C,T)} = \sigma(\text{Conv2}(\text{SiLU}(\mathcal{B}(\text{Conv1}(\text{GAP}_\text{f}(\mathbf{Y})))))) \tag{8}$$

where GAP_f is one-dimensional global average pooling along the frequency dimension.

For the third branch, the attention map between the frequency and temporal dimension (F, T), $\mathbf{S}^{(F,T)} \in \mathbb{R}^{1\times F\times T}$, is computed as follows:

$$\mathbf{S}^{(F,T)} = \sigma(\mathcal{B}(\text{Conv3}([\text{GAP}_\text{c}(\mathbf{Y}) \cdot \text{GMP}(\mathbf{Y})]))) \tag{9}$$

where GAP_c is one-dimensional global average pooling along the channel dimension. GMP means the global max pooling. Conv3 refers to a 7x7 convolution.

Finally, the refined feature $\mathbf{Y}' \in \mathbb{R}^{C\times F\times T}$ by TA can be generated by:

$$\mathbf{Y}' = \mathbf{Y} \otimes \mathbf{S}^{(C,F)} \otimes \mathbf{S}^{(C,T)} \otimes \mathbf{S}^{(F,T)} \tag{10}$$

where $\otimes$ represents the broadcasting multiplication.

4. Experimental Setup

4.1. Dataset and Data Augmentation

We adopt Voxceleb1&2 [26, 27] to validate the proposed DP-Net and TA module in the experiments. The development set of Voxceleb2 is used as training data, which is comprised of 1,092,009 utterances from 5994 speakers. In addition, three data augmentation techniques are employed to increase the diversity of the training data, including online data augmentation [35] with MUSAN [36] and RIR dataset [37], specaugment [38] and speed perturb [39] with 0.9 and 1.1 times speed changes. For testing, the whole Voxceleb1 is utilized as the evaluation data. Performance is reported on three official trial lists: Vox1-O, Vox1-E and Vox1-H.

4.2. System Configuration

The input acoustic features is 80-dimensional filter bank with 25ms windows and 10ms shift. We randomly sample a 200-frame chunk from each utterance during the training process. In addition, AAM-softmax [6] with a margin of 0.2 and a scale of 32 is adopted as the training criterion for all systems. Models are optimized using stochastic gradient descent (SGD) with momentum of 0.9 and weight decay of $1e$-4. The learning rate is controlled by the exponential scheduler decreasing from 0.1 to $1e$-5. During testing, adaptive score normalization (AS-Norm) [40, 41] is adopted to normalize cosine similarity score by setting the imposter cohort as 600. Performance is measured in terms of the equal error rate (EER) and the minimum detection cost function (MinDCF) with the settings of $P_{target} = 0.01$ and $C_{FA} = C_{Miss} = 1$. Specifically, we build four types of systems for comparison, and the configurations of each type are listed as follows:

- Baselines: ResNet18 and ResNet34.
- DPNets: by adopting ResNet18 and Resnet34 as the residual path in the proposed dual path architecture respectively, we can obtain the corresponding DPNet18 and DPNet34.
- Baselines with TA: integrate the TA module into ResNet18 and ResNet34 by inserting it after the batch normalization in each residual bock, which is the same as [30, 31].

Table 2: *EER and MinDCF results of different systems on the Voxceleb1 dataset.*

Architecture	# Params	Voxceleb-O		Voxceleb-E		Voxceleb-H	
		EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF
ResNet18	4.11M	1.48	0.1737	1.52	0.1751	2.72	0.2444
+TA	+0.13M	0.87	0.0803	1.05	0.1199	1.94	0.1914
ResNet34	6.63M	0.96	0.0885	1.01	0.1206	1.86	0.1769
+TA	+0.24M	0.84	0.0796	0.90	0.1055	1.67	0.1609
DPNet18	4.60M	1.27	0.1376	1.31	0.1577	2.36	0.2247
+TA	+0.13M	**0.79**	**0.0794**	**0.99**	**0.1117**	**1.85**	**0.1822**
DPNet34	7.40M	0.81	0.0716	0.89	0.0938	1.65	0.1609
+TA	+0.24M	**0.72**	**0.0658**	**0.74**	**0.0853**	**1.51**	**0.1501**

- DPNets with TA: similarly, the TA module is incorporated into DPNet18 and DPNet34 by inserting it after the batch normalization in each residual block of the residual path as shown in Fig. 1.

5. Results and Analysis

In this section, we first present the results of the proposed DPNets and baseline systems in Table 2. Then the effect of attention modules is analysed in Table 3.

5.1. Results for DPNets

We build DPNet18 and DPNet34 by adopting ResNet18 and ResNet34 as the residual path respectively, where the channel number in the recurrent path is set to 32. We can see that both DPNet18 and DPNet34 outperform the corresponding ResNet18 and ResNet34, which demonstrates the effectiveness of the information exchange between residual and recurrent path. Specifically, for DPNet18, the relative improvements in EER by 14.2%, 13.8%, 13.2% are obtained over the ResNet18 system in the three official trial lists. Additionally, DPNet34 decreases the EERs to 0.81%, 0.89% and 1.65% on Vox1-O, Vox1-E and Vox1-H respectively. It reveals that the introduction of the recurrent path along depth into CNN is beneficial to feature re-usage and re-exploitation. The success of the proposed framework can be attributed to the fact that information from different layers is accumulated and exchanged between the residual and recurrent path. Rather than a simple combination of two paths, dynamic interaction is vital to improve the representation capabilities of the SV system.

5.2. The Effect of Attention Modules

5.2.1. TA Module

TA module can be easily integrated into both ResNet and DPNet, which results in significant improvements with negligible computational overhead. For ResNet, the averagely relative improvements in EER by 26.8%, 20.8%, 19.4% are obtained over the ResNet18 and ResNet34 system in the three official trial lists. Similarly, DPNet18-TA and DPNet34-TA averagely decrease the EERs on Vox1-O, Vox1-E and Vox1-H by 24.4%, 20.6% and 15.1% respectively. It demonstrates the effectiveness and importance of modeling cross-dimension attentions for speaker verification. Moreover, it is noteworthy that the combination of DPNet and TA module yields the best performance, which indicates that the functionalities of the recurrent path and TA module are complementary.

Table 3: *The effect of different attention modules.*

System	# Params	Vox1-O	Vox1-E	Vox1-H
ResNet18	4.11M	1.48	1.52	2.72
+SE	+0.09M	1.39	1.50	2.66
+DCFT	+0.13M	1.29	1.43	2.63
+SimAM	+0	1.51	1.59	2.80
+TA(ours)	+0.13M	**0.87**	**1.05**	**1.94**

5.2.2. Comparison with Previous Attention Mechanisms

To validate the superiority of our proposed TA module over previous methods, we adopt ResNet18 as the baseline and reimplement commonly-used attention mechanisms in SV task such as SE [28], DTCF [30] and SimAM [31]. As shown in Table 3, the effect of SE module is very limited, which reveals that only modeling channel-wise dependencies is insufficient for SV task. DTCF attempts to assemble the temporal and frequency information into the channel-wise attention. However, the improvements are still not significant. For SimAM, although it introduces no additional parameters, the performance becomes even worse than the baseline. Different from the above methods, our TA module interactively models the inter-dependencies between channel and frequency, channel and temporal, frequency and temporal separately, which outperforms all listed methods by a large margin. This demonstrates the importance of capturing cross dimension interactions for SV task.

6. Conclusions

In this paper, we introduce a novel dual path embedding learning framework for speaker verification. By accumulating and exchanging information between two paths, enhanced features can be learned to improve the representation capabilities of SV system. In addition, an efficient triplet attention module is proposed to model cross-dimension attentions. Experiments on Voxceleb dataset show that the proposed DPNets can ourperform the corresponding ResNets. And further improvements can be obtained when combining with triplet attention module.

7. Acknowledgement

This work was supported in part by China NSFC projects under Grants 62122050 and 62071288, and in part by Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0102.

8. References

[1] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[2] S. Ioffe, "Probabilistic linear discriminant analysis," in *ECCV*, 2006, pp. 531–542.

[3] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *ICASSP*, 2014, pp. 4052–4056.

[4] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[5] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *INTERSPEECH*, 2017, pp. 999–1003.

[6] Y. Liu, L. He, and J. Liu, "Large margin softmax loss for speaker verification," in *INTERSPEECH*, 2019, pp. 2873–2877.

[7] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: robust dnn embeddings for speaker recognition," in *ICASSP*, 2018, pp. 5329–5333.

[8] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *ICASSP*, 2019, pp. 5796–5800.

[9] D. Garcia-Romero, A. McCree, D. Snyder, and G. Sell, "Jhu-hltcoe system for the voxsrc speaker recognition challenge," in *ICASSP*, 2020, pp. 7559–7563.

[10] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *INTERSPEECH*, 2020, pp. 3830–3834.

[11] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "But system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.

[12] B. Liu, H. Wang, Z. Chen, S. Wang, and Y. Qian, "Self-knowledge distillation via feature enhancement for speaker verification," in *ICASSP*, 2022.

[13] P. Safari, M. India, and J. Hernando, "Self-attention encoding and pooling for speaker recognition," in *INTERSPEECH*, 2020, pp. 941–945.

[14] B. Han, Z. Chen, and Y. Qian, "Local information modeling with self-attention for speaker verification," in *ICASSP*, 2022.

[15] B. Han, Z. Chen, B. Liu, and Y. Qian, "Mlp-svnet : a multi-layer perceptrons based network for speaker verification," in *ICASSP*, 2022.

[16] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015, pp. 3214–3218.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: transformers for image recognition at scale," in *ICLR*, 2021.

[20] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *NIPS*, 2017, pp. 4470–4478.

[21] J. Moniz and C. Pal, "Convolutional residual memory networks," *arXiv preprint arXiv:1606.05262*, 2016.

[22] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015, pp. 2625–2634.

[23] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *NIPS*, 2015, pp. 802–810.

[24] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, "Conformer: local features coupling global representations for visual recognition," in *ICCV*, 2021, pp. 367–376.

[25] N. Park and S. Kim, "How do vision transformers work?" in *ICLR*, 2022.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017, pp. 2616–2620.

[27] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: deep speaker recognition," in *INTERSPEECH*, 2018, pp. 1086–1090.

[28] J. Zhou, T. Jiang, Z. Li, L. Li, and Q. Hong, "Deep speaker embedding extraction with channel-wise feature responses and additive supervision softmax loss function," in *INTERSPEECH*, 2019, pp. 2883–2887.

[29] S. Yadav and A. Rai, "Frequency and temporal convolutional attention for text-independent speaker recognition," in *ICASSP*, 2020, pp. 6789–6793.

[30] L. Zhang, Q. Wang, and L. Xie, "Duality temporal-channel-frequency attention enhanced speaker representation learning," *arXiv preprint arXiv:2110.06565*, 2021.

[31] X. Qin, N. Li, C. Weng, D. Su, and M. Li, "Simple attention module based speaker verification with iterative noisy label detection," *arXiv preprint arXiv:2110.06534*, 2021.

[32] J. Zhao, Y. Fang, and G. Li, "Recurrence along depth: deep convolutional neural networks with recurrent layer aggregation," in *NIPS*, 2021, pp. 10 627–10 640.

[33] S. Wang, Y. Yang, Y. Qian, and K. Yu, "Revisiting the statistics pooling layer in deep speaker embedding learning," in *ISCSLP*, 2021, pp. 1–5.

[34] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *arXiv preprint arXiv:1702.03118*, 2017.

[35] W. Cai, J. Chen, J. Zhang, and M. Li, "On-the-fly data loader and utterance-level aggregation for speaker and language recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 28, pp. 1038–1051, 2020.

[36] D. Snyder, G. Chen, and D. Povey, "Musan: a music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[37] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*, 2017, pp. 5220–5224.

[38] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: a simple data augmentation method for automatic speech recognition," in *INTERSPEECH*, 2019, pp. 2613–2617.

[39] W. Wang, D. Cai, X. Qin, and M. Li, "The dku-dukeece systems for voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2010.12731*, 2020.

[40] Z. N. Karam, W. M. Campbell, and N. Dehak, "Towards reduced false-alarms using cohorts," in *ICASSP*, 2011, pp. 4512–4515.

[41] S. Cumani, P. Batzu, D. Colibro, C. Vair, P. Laface, and V. Vasilakakis, "Comparison of speaker recognition approaches for real applications," in *INTERSPEECH*, 2011, pp. 2365–2368.